C. R. Mecanique 344 (2016) 95–101

Contents lists available at ScienceDirect

Comptes Rendus Mecanique

www.sciencedirect.com

Higher-order continuation for the determination of robot workspace boundaries

Continuation de haut degré pour la détermination des frontières de l'espace de travail d'un robot

Gauthier Hentz *, Isabelle Charpentier, Pierre Renaud

ICube – Laboratoire des sciences de l'ingénieur, de l'informatique et de l'imagerie, Université de Strasbourg, CNRS, INSA Strasbourg, 300, bd Sébastien-Brant, CS 10413, 67412 Illkirch, France

ARTICLE INFO

Article history:
Received 1 September 2015
Accepted 10 October 2015
Available online 11 November 2015

Keywords:
Mechanisms
Workspace boundaries
Extended system of equations
Automatic differentiation
Taylor series
Diamant

Mots-clés :
Mécanismes
Frontières de l'espace de travail
Système d'équations étendu
Différentiation Automatique
Séries de Taylor
Diamant

ABSTRACT

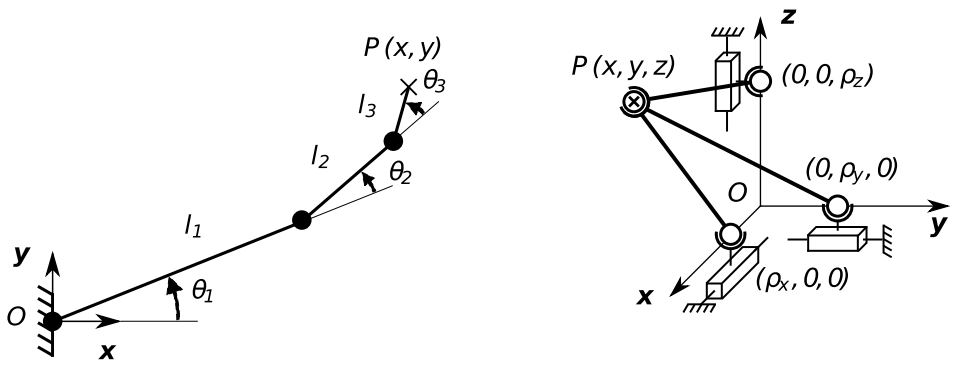
In the medical and surgical fields, robotics may be of great interest for safer and more accurate procedures. Space constraints for a robotic assistant are however strict. Therefore, roboticists study non-conventional mechanisms with advantageous size/workspace ratios. The determination of mechanism workspace, and primarily its boundaries, is thus of major importance. This Note builds on boundary equation definition, continuation and automatic differentiation to propose a general, accurate, fast and automated method for the determination of mechanism workspace. The method is illustrated with a planar RRR mechanism and a three-dimensional Orthoglide parallel mechanism.

RÉSUMÉ

Dans le contexte médico-chirurgical, la robotique peut être d'un grand intérêt pour des procédures plus sûres et plus précises. Les contraintes de compacité pour un dispositif robotique sont cependant fortes. En conséquence, les roboticiens étudient des mécanismes non conventionnels présentant des ratios taille/espace de travail avantageux. La détermination de l'espace de travail, et en premier lieu de ses frontières, est donc primordiale. Cette Note repose sur la définition des équations de frontière, la continuation et la différentiation automatique pour proposer une méthode de détermination de l'espace de travail des mécanismes générique, précise, rapide et automatisée. La méthode est illustrée sur un mécanisme RRR plan et un mécanisme parallèle tridimensionnel, l'Orthoglide.

* Corresponding author.
E-mail address: gauthier.hentz@icube.unistra.fr (G. Hentz).

http://dx.doi.org/10.1016/j.crme.2015.10.001



(a) Planar RRR mechanism. (b) Orthoglide mechanism, adapted from [10].

Fig. 1. Kinematic schemes of the evaluated mechanisms.

1. Introduction

In the medical and surgical fields, a robotic assistance may be of great interest for safer and more accurate procedures. Space is however strongly limited in the operating room or in imaging facilities such as CT and MRI scanners. Roboticists therefore study and design non-conventional mechanisms with advantageous ratios between their size and the reachable positions of their end-effector, *i.e.* their workspace. Developing a general, accurate, fast and automated method for the determination of mechanism workspaces, in particular its boundaries, is thus of prime importance.

Several methods are reported in the literature [1–3], but only a few devise general frameworks [4–6,3]. On the one hand, workspace boundaries may be described by applying the implicit function theorem to the mechanism equations as discussed by Litvin [4] in the early eighties. Their computation may be carried out using a Newton–Raphson continuation method to follow one-dimensional curves on the workspace boundaries as proposed by Haug et al. [5] in the nineties. This implementation requires a second order differentiation of the mechanism equations since the Jacobian resulting from the application of the implicit function theorem is to be differentiated for the Newton–Raphson method. The differentiation steps and the one-dimensional path calculation have constituted limitations to its dissemination. On the other hand, Merlet et al. [6] introduce an interval analysis method for the workspace analysis. It consists in an iterative bisection method of an initial box containing the workspace to determine inner and bordering sub-boxes of the workspace. Bohigas et al. [3] combine elements of both approaches in a so-called branch-and-prune method. The interval analysis method is then applied to the mechanism workspace boundary equations [4] written in a quadratic form to enable a faster shrinking of solution boxes. In both cases [6,3], the determination of the workspace boundaries with sufficiently small sub-boxes requires an extensive computational effort, in particular for high dimensional workspaces.

As demonstrated in earlier works [7,8], Automatic Differentiation (AD) is a powerful general approach for the implementation of a higher-order continuation method. This Note is based upon the workspace boundary formulations [4,5] and the Diamant framework [9] to propose a general, efficient and automated higher-order continuation method that allows for the continuous representation of workspace boundaries. The paper is organized as follows. Workspace definitions are presented in Section 2 and exemplified on the classical RRR mechanism [5,1,3]. The general implementation – construction of the boundary equations of the mechanism and higher order continuation – is discussed in Section 3. Numerical results obtained for the two-dimensional RRR mechanism and a three-dimensional mechanism, the Orthoglide [10], are reported in Section 4.

2. Formulation of workspace boundaries

In order to introduce the formulation of workspace boundaries, let us first consider the well-known RRR mechanism, see Fig. 1(a). This serial mechanism is composed of three revolute joints linking three bars of length l_1, l_2, l_3, respectively. The first bar rotates about the base by an angle θ_1, the second bar rotates with respect to the first one by an angle θ_2, and so forth. The position of the end-effector P is denoted by (x, y).

The vector $(\theta_1, \theta_2, \theta_3)$ forms a set of so-called generalized coordinates that is sufficient to describe the configuration of the mechanism in a unique manner. The behavior of the mechanism can be modeled by nonlinear kinematic constraint equations issued from loop closure relationships, that is:

$$\begin{cases} x - (l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2) + l_3 \cos(\theta_1 + \theta_2 + \theta_3)) = 0 \\ y - (l_1 \sin(\theta_1) + l_2 \sin(\theta_1 + \theta_2) + l_3 \sin(\theta_1 + \theta_2 + \theta_3)) = 0 \end{cases} \tag{1}$$

Joint angular ranges are usually restricted for technological reasons, assuming $\theta_i \in [-\pi/3; \pi/3]$ for $i \in \{1, 2, 3\}$ for instance. Equations (1) are reformulated using the input vector $\mathbf{v} = (v_1, v_2, v_3)$ with $\theta_i = \pi/3.\sin(v_i)$, $i \in \{1, 2, 3\}$ to take into account

these restrictions. The mechanism equations are then written in the form of the general nonlinear residual problem $\mathbf{R}^{\mathcal{W}}$ such that

$$\mathbf{R}^{\mathcal{W}}(\mathbf{u},\mathbf{v})=0 \tag{2}$$

linking the input vector $\mathbf{v}$ with the vector of output coordinates $\mathbf{u}=(x,y)$. In the general case $\mathbf{R}^{\mathcal{W}}$ is a set of n_e residual equations that relates the output coordinates $\mathbf{u}\in\mathbb{R}^{n_u}$ to the input coordinates $\mathbf{v}\in\mathbb{R}^{n_v}$.

From a robotic point of view, the most meaningful information on a mechanism is the reachable workspace $\mathcal{W}$ defined as

$$\mathcal{W}=\left\{\mathbf{u}\mid\exists\,\mathbf{v}\text{ such that }\mathbf{R}^{\mathcal{W}}(\mathbf{u},\mathbf{v})=0\right\} \tag{3}$$

and determined by the mechanism geometry and the joint constraints. The workspace is therefore a manifold of very general structure, the boundaries of which correspond to the singular configurations of the mechanism and are of difficult calculation. Following [4,5], the workspace boundary $\partial\mathcal{W}$ is defined by

$$\partial\mathcal{W}=\left\{\mathbf{u}\in\mathcal{W}\mid\exists\,(\xi,\mathbf{v})\text{ such that }\left(\mathbf{R}^{\mathcal{W}}_{\mathbf{v}}\right)^{\mathsf{T}}(\mathbf{u},\mathbf{v}).\xi=\mathbf{0}\text{ and }\xi^{\mathsf{T}}.\xi=1\right\} \tag{4}$$

where $\mathbf{R}^{\mathcal{W}}_{\mathbf{v}}$ is the Jacobian of $\mathbf{R}^{\mathcal{W}}$ computed with respect to $\mathbf{v}$, the transpose of which is denoted by $(\mathbf{R}^{\mathcal{W}}_{\mathbf{v}})^{\mathsf{T}}$. The existence of a unit vector $\xi\in\mathbb{R}^{n_e}$ implies the row rank deficiency of $\mathbf{R}^{\mathcal{W}}_{\mathbf{v}}$. Vector ξ points towards the impeded direction of motion. This yields the following boundary residual problem:

$$\mathbf{R}^{\partial\mathcal{W}}(\mathbf{u},\mathbf{v},\xi)=\begin{bmatrix}\mathbf{R}^{\mathcal{W}}(\mathbf{u},\mathbf{v})\\ \left(\mathbf{R}^{\mathcal{W}}_{\mathbf{v}}\right)^{\mathsf{T}}(\mathbf{u},\mathbf{v}).\xi\\ \xi^{\mathsf{T}}.\xi-1\end{bmatrix}=0 \tag{5}$$

The system (5) is composed of n_e+n_v+1 nonlinear equations with $n=n_e+n_v+n_u$ unknowns. The dimension of $\partial\mathcal{W}$ is thus equal to (n_u-1). In other words, the dimension of the workspace boundaries is one less than the workspace dimension. The workspace boundary computation may then be carried out by a continuation method. The interested reader is referred to the early work of Seydel [11,12] for a mathematical presentation of such extended systems of equations.

This Note broadens the first order continuation studies [5,12] by adapting the general and automated higher-order continuation framework Diamant [9] to the workspace boundary determination issue.

3. Full AD-based continuation process

The determination of solutions for nonlinear under-determined residual equations is a frequent issue in mechanics. Continuation is then a common tool, with the example in robotics of path planning through homotopy, see [13] and references therein, and workspace boundary computation [5]. Haug's method [5] has raised attention in robotics, but despite accurate calculations and some use of AD, the Jacobian calculations and the discrete one-dimensional continuation issues have indeed limited the reuse of the method.

This Note proposes a full AD-based response to these challenges through the differentiation of the computer codes [14] related to $\mathbf{R}^{\mathcal{W}}(\mathbf{u},\mathbf{v})$ and $\mathbf{R}^{\partial\mathcal{W}}(\mathbf{u},\mathbf{v},\xi)$ with respect to their input variables. Firstly, Subsection 3.1, some differentiation by source transformation [15] of $\mathbf{R}^{\mathcal{W}}(\mathbf{u},\mathbf{v})$ allows for the construction of the code of the boundary residual problem $\mathbf{R}^{\partial\mathcal{W}}(\mathbf{u},\mathbf{v},\xi)$. Secondly, Subsection 3.2, operator overloading [9] is used on $\mathbf{R}^{\partial\mathcal{W}}(\mathbf{u},\mathbf{v},\xi)$ to carry out the derivative computations and the higher-order continuation within Diamant.

3.1. Construction of the boundary residual problem $\mathbf{R}^{\partial\mathcal{W}}(\mathbf{u},\mathbf{v},\xi)$

Jacobian calculation may be carried out in a very simple manner using an AD tool, or even by hand. Given a computer code, it suffices to differentiate statements and to evaluate them in the required directions of perturbation. In the present case, the residual $\mathbf{R}^{\mathcal{W}}(\mathbf{u},\mathbf{v},\xi)$ is differentiated with respect to the input coordinates $\mathbf{v}$ to get $\mathbf{R}^{\mathcal{W}}_{\mathbf{v}}(\mathbf{u},\mathbf{v},\xi)$. In earlier works, Haug et al. [5] suggested to use ADOL-C, a higher-order overloading AD tool, for the evaluation of the Jacobian $\mathbf{R}^{\mathcal{W}}_{\mathbf{v}}(\mathbf{u},\mathbf{v},\xi)$ defining the boundaries.

The continuation method also requires some Hessian calculation for the evaluation of the Jacobian intervening in the Newton–Raphson iterations. This Hessian may be evaluated using ADOL-C choosing the right directions of perturbation, but it is not a trivial task for any mechanism.

Source transformation AD tools are more intuitive tools for Jacobian calculation as they allow for the generation of tangent linear codes containing the tangent linear statements corresponding to $\mathbf{R}^{\mathcal{W}}_{\mathbf{v}}(\mathbf{u},\mathbf{v},\xi)$ while replicating the statement of $\mathbf{R}^{\mathcal{W}}(\mathbf{u},\mathbf{v})$. For any mechanism, the equations of the boundary residual problem may be formally obtained from $\mathbf{R}^{\mathcal{W}}(\mathbf{u},\mathbf{v})$ by means of the following three-step process:

$$\mathbf{R}^{\mathcal{W}}(\mathbf{u},\mathbf{v}) \xrightarrow[\text{system}]{\text{Augment}} \begin{pmatrix} \mathbf{R}^{\mathcal{W}}(\mathbf{u},\mathbf{v}) \\ \left(\mathbf{R}^{\mathcal{W}}\right)^{\mathsf{T}}(\mathbf{u},\mathbf{v}).\xi \\ \xi^{\mathsf{T}}.\xi - 1 \end{pmatrix} \xrightarrow[\text{w.r.t. } \mathbf{v}]{\text{Differentiate}} \begin{pmatrix} \mathbf{R}^{\mathcal{W}}(\mathbf{u},\mathbf{v}) \\ \mathbf{R}^{\mathcal{W}}_{\mathbf{v}}(\mathbf{u},\mathbf{v}) \\ \left(\mathbf{R}^{\mathcal{W}}\right)^{\mathsf{T}}(\mathbf{u},\mathbf{v}).\xi \\ \left(\mathbf{R}^{\mathcal{W}}_{\mathbf{v}}\right)^{\mathsf{T}}(\mathbf{u},\mathbf{v}).\xi \\ \xi^{\mathsf{T}}.\xi - 1 \end{pmatrix} \xrightarrow{\textit{Reduce}} \begin{pmatrix} \mathbf{R}^{\mathcal{W}}(\mathbf{u},\mathbf{v}) \\ \left(\mathbf{R}^{\mathcal{W}}_{\mathbf{v}}\right)^{\mathsf{T}}(\mathbf{u},\mathbf{v}).\xi \\ \xi^{\mathsf{T}}.\xi - 1 \end{pmatrix} \tag{6}$$

Among the existing AD tools, we choose Tapenade [15] for its applicability to Fortran codes and its free interactive web interface. Moreover, its "multi-directional tangent (linear) mode" of differentiation gives access to the Jacobian calculation through the simultaneous evaluation of the tangent linear derivatives in the canonical basis of $\mathbb{R}^{n_v}$. By convention, tangent linear routines and variables are suffixed with `_d` and `d`, respectively.

Following (6), the proposed method for the construction of the boundary residual problem (5) consists in:

1. defining the continuous residual equations $\mathbf{R}^{\mathcal{W}}(\mathbf{u},\mathbf{v})$ and the input/output coordinates $\mathbf{u}$ and $\mathbf{v}$ of the mechanism under study,
2. implementing the extended system $(\mathbf{R}^{\mathcal{W}}(\mathbf{u},\mathbf{v}), \left(\mathbf{R}^{\mathcal{W}}\right)^{\mathsf{T}}(\mathbf{u},\mathbf{v}).\xi, \xi^{\mathsf{T}}.\xi - 1)$ into a subroutine `ResidualW(u,v,Xi,RW, RWtXi,nXi)`. Input arguments are `u`, `v` and `Xi` while output arguments for the different functions are denoted by `RW`, `RWtXi` and `nXi`, respectively,
3. differentiating `ResidualW(u,v,Xi,RW,RWtXi,nXi)` with respect to the argument `v` using Tapenade in "multi-directional tangent (linear) mode" to get the subroutine `ResidualW_d(u,v,vd,RW,RWd,RWtXi,RWtXid,nXi)`.

At evaluation time, the tangent variable `vd` is set equal to the $n_v \times n_v$ identity matrix which describes the canonical basis of $\mathbb{R}^{n_v}$. The subroutine `ResidualW_d` contains the necessary information to the evaluation of the boundary residual system (5). In other words, it may be used in the Diamant framework for the computation of continuous solution branches on the workspace boundary.

3.2. Higher-order continuation

The constraint equations (2) and the boundary problem (5) are both defined as nonlinear residual problems, and can be both solved using a continuation method. While the load parameter λ is usually chosen as the continuation parameter in structural mechanics, this is not always possible nor pertinent in robotics. In that case, one uses one of the input/output coordinates as a continuation parameter.

The higher-order continuation framework Diamant [7,9] combines the Asymptotic Numerical Method [16] with AD [14] to compute the solution to the general under-determined nonlinear residual problem

$$\mathcal{R}(\mathbf{U}(a)) = 0 \tag{7}$$

where $\mathbf{U}(a)$ denotes the unknown state vector and a is a path parameter satisfying, for instance, the pseudo-arclength equation

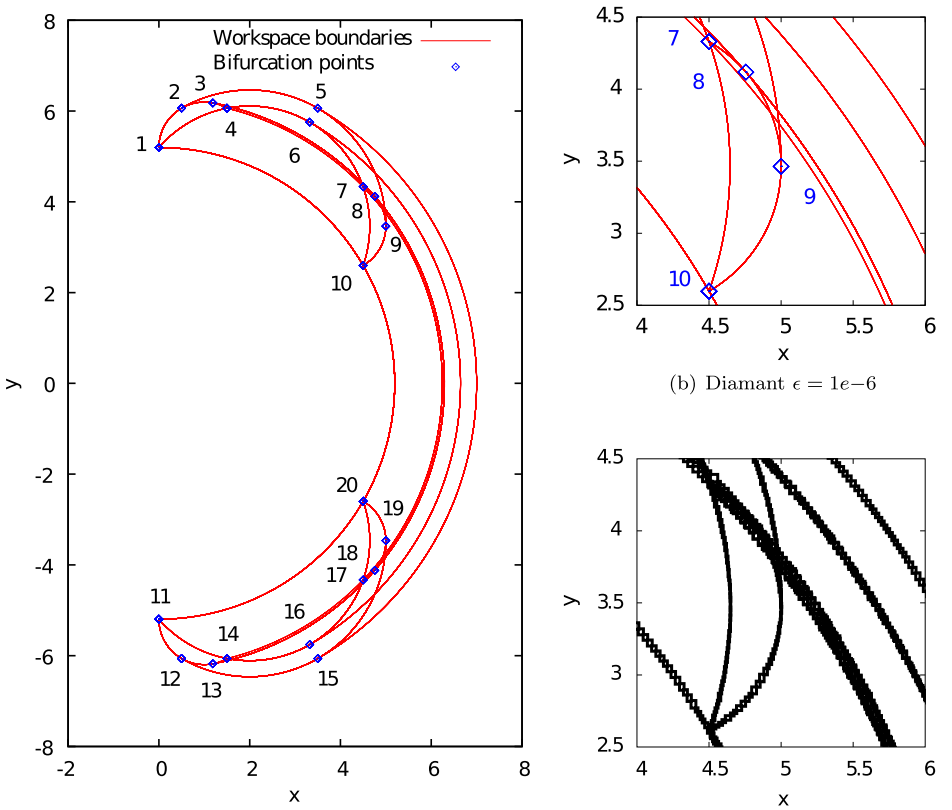
$$a = \left\langle \mathbf{U}(a) - \mathbf{U}(0), \frac{\partial \mathbf{U}}{\partial a}(0) \right\rangle \tag{8}$$

to close the under-determined residual problem. Under analyticity assumptions, continuous solution branches $\mathbf{U}(a)$ may be approximated as Taylor expansions $\mathbf{U}(a) = \sum_{k=0}^{K} a^k \mathbf{U}_k$ where K is the truncation order and $\mathbf{U}_k = \frac{1}{k!}\frac{\partial^k \mathbf{U}}{\partial a^k}(0)$ is the Taylor coefficient of $\mathbf{U}$ at order k. The nonlinear residual problem (7) may be then turned into the following iterative sequence of K linear systems:

$$\begin{cases} \{\mathcal{R}_1\}\mathbf{U}_1 = -\left\{\mathcal{R}_{1|\mathbf{U}_1=\mathbf{0}}\right\}, & \langle \mathbf{U}_1, \mathbf{U}_1 \rangle = 1 \\ \{\mathcal{R}_1\}\mathbf{U}_k = -\left\{\mathcal{R}_{k|\mathbf{U}_k=\mathbf{0}}\right\}, & \langle \mathbf{U}_k, \mathbf{U}_1 \rangle = 0, \quad \text{for } k = 2, .., K \end{cases} \tag{9}$$

Therein, the Jacobian $\{\mathcal{R}_1\}$ is the same over the orders and $\{\mathcal{R}_{k|\mathbf{U}_k=\mathbf{0}}\}$ represents the contribution of higher-order nonlinear terms. Both terms may be computed using an operator overloading AD. This signifies the solution for the iterative sequence (9) may be fully automated. The interested reader is referred to [8,9] for details.

Under analyticity assumptions, the workspace boundaries may be thus computed from (5) using $\mathcal{R} = \mathbf{R}^{\partial\mathcal{W}}$ and $\mathbf{U} = (\mathbf{u}, \mathbf{v}, \xi)$. Moreover, the proposed workspace boundary determination method inherits the generality, accuracy and efficiency from the Diamant framework. Computed as Taylor expansions, one-dimensional continuous solution branches can be used for surface reconstruction of higher-dimensional workspace boundaries.



(a) RRR reachable workspace

(b) Diamant $\epsilon = 1e{-}6$

(c) Branch-and-prune $\sigma = 5e{-}2$

Fig. 2. (a) Boundaries of the RRR mechanism reachable workspace. (b) Zoom on the solution branches. (c) Zoom on the boundaries computed with the branch-and-prune method implemented in [21].

4. Numerical results

In the following, the abilities and efficiency of the proposed method are assessed considering the two-dimensional workspace of the RRR mechanism. The generality of the method is then evaluated considering the three-dimensional Orthoglide mechanism. The method is implemented in Fortran 90 and run on an Intel Core i7, 2.7 GHz, 16 GB RAM. Residuals are computed with an accuracy of $\epsilon = 1\text{e}{-}6$.

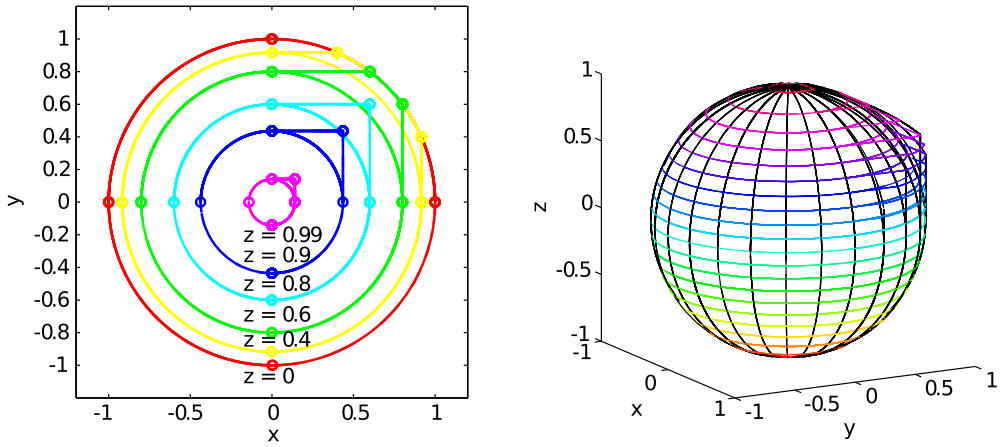
4.1. Workspace boundaries of the RRR mechanism

The planar RRR mechanism has a two-dimensional worskpace. Its boundaries are one-dimensional curves that can be followed by continuation. The mechanism dimensions are set to $(l_1, l_2, l_3) = (4, 2, 1)$ and the simulation is run from the horizontally extended configuration $(\mathbf{u}, \mathbf{v}) = (l_1 + l_2 + l_3, 0, 0, 0, 0)$ with the normal vector $\xi = (1, 0)$ and a truncation order of $K = 20$. The solution branches $(\mathbf{u}(a), \mathbf{v}(a), \xi(a))$ are projected on the output coordinate space to plot the computed workspace boundary $\partial\mathcal{W}$, see Fig. 2(a). The workspace is bounded by the external curves, while the interior curves are loci of output motion restrictions. As already noticed in [5], the projected bifurcation diagram comprises 20 continuous solution branches linking the numbered bifurcation points. These are detected by monitoring a change of sign for the Jacobian determinant [17]. The interested reader may refer to [18,19] for higher order bifurcation indicators.

The RRR mechanism has three input coordinates $\mathbf{v}$ and two output coordinates $\mathbf{u}$. The nonlinear kinematic constraint equations are surjective. For a prescribed position $\mathbf{u} \in R^{\mathcal{W}}(\mathbf{u}, \mathbf{v})$, their solutions are input coordinate sets called inverse kinematics solutions (IKS). Branches related to different IKS may overlap (branches 1–10–20 and 10–20–11, for instance) or intersect at bifurcation points. Although it seems possible on the diagram, switching continuously from one branch to another one is not always possible. This is the case at the bifurcation point number 10 where the “elbow-up” IKS branch issued from $\mathbf{v} = (-\pi/2, -\pi/2, -\pi/2)$ runs from the points 11 to 10 (no bifurcation at point 20), then from points 10 to 9, while the “elbow-down” IKS branch issued from $\mathbf{v} = (\pi/2, \pi/2, \pi/2)$ runs from the points 1 to 20 (no bifurcation at point 10). Branching and switching from one IKS to another are carried out by applying a random perturbation to the solution points close to a bifurcation [20]. A systematic branch bifurcation method based on [19] will be considered in a future work for a deeper understanding of the path following along mechanism workspace boundaries.

Table 1
Computation time of the RRR workspace boundaries with given precision threshold ϵ for the Diamant and the branch-and-prune methods.

Method	Computation time	Accuracy ϵ
Branch-and-prune	1 min 15 s	5e−2
Branch-and-prune	4 min 31 s	1e−2
Branch-and-prune	7 min 36 s	5e−3
Branch-and-prune	196 min 42 s	1e−4
Diamant	7 s	1e−6



(a) Iso-altitude map (b) Isometric view

Fig. 3. (a) Slices of the Orthoglide workspace boundaries with $z = \{0, 0.4, 0.6, 0.8, 0.9, 0.99\}$. (b) 3D representation of the Orthoglide mechanism workspace boundaries.

Fig. 2 and Table 1 allow for a comparison between the proposed continuation method and the branch-and-prune method [3] implemented in CUIK++ software [21]. First, see Fig. 2(c), CUIK++ bounds the workspace singularities into small boxes with a prescribed size σ. Second, see Table 1, the higher the accuracy, the longer the computational time for CUIK++. The continuation method is much faster and more accurate than CUIK++. Third, branching points are not accurately determined.

4.2. Orthoglide workspace boundaries

The isotropic closed-chain Orthoglide mechanism [10], Fig. 1(b), is chosen as a second illustration. This spatial mechanism presents valuable kinematics properties, in particular with the existence of an isotropic configuration around which the transmission factor between input and output velocities is close to unity. In a simplified modeling, it consists of three identical legs of length l. Each leg is connected to the end-effector P by a passive spherical joint at one end, and to an actuated prismatic joint through a passive spherical joint at the other end. The input coordinates $\mathbf{v} = (\rho_x, \rho_y, \rho_z)$ are the prismatic joint variables and the output coordinates $\mathbf{u} = (x, y, z)$ describe the position of P. The modeling and parameterization introduced in [10] are converted into a nonlinear residual problem satisfying (2).

The continuation process can be used to compute boundaries of workspaces of dimension larger than 2 [5]. For a three-dimensional space, a parameterization that links the output coordinates is added to allow for the computation of solution branches. A possible choice is (*i*) to convert the Cartesian output coordinates (x, y, z) into cylindrical coordinates (r, ϕ, z), (*ii*) to fix either z or ϕ, then (*iii*) to carry out a continuation with respect ϕ or z, respectively. The process is repeated for different values of z or ϕ, respectively, to get a set of isolines, see Fig. 3(a).

These curves are assembled in Fig. 3(b) to build the workspace boundaries. Fig. 3(b) lets appear a sphere of radius l and a convex surface limited by configurations in which at least one leg is orthogonal to the axis defining its prismatic joint. The shape of the convex surface, geometrically bounded by three cylinders of radius l, may be better observed by means of the projection on the (x, y)-plane in Fig. 3(a). Bifurcation points represent loci of multiple IKS. This representation agrees with the geometrical description elaborated in [10].

As a parallel mechanism, the Orthoglide also exhibits parallel singularities corresponding to the singularities of the Jacobian $\mathbf{R}_{\mathbf{u}}^{\mathcal{W}}$. Determination of parallel singularity surfaces [10] should be possible as well with the proposed continuation method by differentiating the mechanism equations with respect to the output coordinates $\mathbf{u}$ and substituting $\mathbf{R}_{\mathbf{v}}^{\mathcal{W}}$ by $\mathbf{R}_{\mathbf{u}}^{\mathcal{W}}$ in Eq. (5).

5. Conclusions

Based on automatic differentiation and Diamant, this Note describes a general and automated framework for the determination of mechanism workspace boundaries. This framework inherits abilities in terms of accuracy and computational time efficiency from the underlying higher-order continuation method. The process designed for the construction of the boundary residual problem is of particular interest as it allows for the automation of such kind of studies, including extended systems of equations [11,12]. The user has only to provide mechanism equations.

As illustrated on the Orthoglide mechanism, the solution branches computed as Taylor expansions allow for the reconstruction of the boundaries of a three-dimensional workspace. Having in mind the design of medical robotic assistants, future works will exploit the sensitivity computations already implemented in the Diamant framework [22] to study some of the mechanism properties close to singular configurations.

References

[1] J.A. Snyman, L.J. du Plessis, J. Duffy, An optimization approach to the determination of the boundaries of manipulator workspaces, J. Mech. Des. 122 (1998) 447–456.
[2] J. Yang, K. Abdel-Malek, Y. Zhang, On the workspace boundary determination of serial manipulators with non-unilateral constraints, Robot. Comput.-Integr. Manuf. 24 (2008) 60–76.
[3] O. Bohigas, M. Manubens, L. Ros, A complete method for workspace boundary determination on general structure manipulators, IEEE Trans. Robot. 28 (2012) 993–1006.
[4] F.L. Litvin, Application of theorem of implicit function system existence for analysis and synthesis of linkages, Mech. Mach. Theory 15 (1980) 115–125.
[5] E.J. Haug, C.-M. Luh, F.A. Adkins, J.-Y. Wang, Numerical algorithms for mapping boundaries of manipulator workspaces, J. Mech. Des., Trans. ASME 118 (1996) 228–234.
[6] J.-P. Merlet, Determination of 6D-workspaces of Gough-type parallel manipulator and comparison between different geometries, Int. J. Robot. Res. 18 (1999) 902–916.
[7] I. Charpentier, M. Potier-Ferry, Automatic differentiation of the asymptotic numerical method: the Diamant approach, C. R. Mecanique 336 (2008) 336–340.
[8] Y. Koutsawa, I. Charpentier, E.M. Daya, M. Cherkaoui, A generic approach for the solution of nonlinear residual equations. Part I: the Diamant toolbox, Comput. Methods Appl. Mech. Eng. 198 (2008) 572–577.
[9] I. Charpentier, On higher-order differentiation in nonlinear mechanics, Optim. Methods Softw. 27 (2012) 221–232.
[10] A. Pashkevich, D. Chablat, P. Wenger, Kinematics and workspace analysis of a three-axis parallel manipulator: the Orthoglide, Robotica 24 (2006) 39–49.
[11] R. Seydel, Numerical computation of branch points in nonlinear equation, Numer. Math. 33 (1979) 339–352.
[12] R. Seydel, Practical Bifurcation and Stability Analysis, 3rd edition, Springer, 2009.
[13] A.J. Sommese, J. Verschelde, C.W. Wampler, Advances in polynomial continuation for solving problems in kinematics, J. Mech. Des. 126 (2004) 262–268.
[14] A. Griewank, A. Walther, Evaluating Derivatives, Principles and Techniques of Algorithmic Differentiation, Society for Industrial and Applied Mathematics, 2008.
[15] L. Hascoet, V. Pascual, The Tapenade automatic differentiation tool: principles, model, and specification, ACM Trans. Math. Softw. 39 (2013) 20:1–20:43.
[16] B. Cochelin, N. Damil, M. Potier-Ferry, Méthode asymptotique numérique, Eur. J. Comput. Mech. 17 (2008) 553–554.
[17] H. Keller, Lectures on Numerical Methods in Bifurcation Problems, Springer-Verlag, Berlin, 1987.
[18] Y. Guevel, H. Boutyour, J.M. Cadou, Automatic detection and branch switching methods for steady bifurcation in fluid mechanics, J. Comput. Phys. 230 (2011) 3614–3629.
[19] B. Cochelin, M. Medale, Power series analysis as a major breakthrough to improve the efficiency of asymptotic numerical method in the vicinity of bifurcations, J. Comput. Phys. 236 (2013) 594–607.
[20] E.L. Allgower, K. Georg, Numerical Continuation Methods, vol. 33, Springer-Verlag, Berlin, 1990.
[21] J.M. Porta, L. Ros, O. Bohigas, M. Manubens, C. Rosales, L. Jaillet, The CUIK suite: analyzing the motion of closed-chain multibody systems, IEEE Robot. Autom. Mag. 21 (2014) 105–114.
[22] I. Charpentier, Sensitivity of solutions computed through the asymptotic numerical method, C. R. Mecanique 336 (2008) 788–793.